DNSSEC
From a protocol bug to a security advantage

Lutz Donnerhacke

db089309: 1c1c 6311 ef09 d819 e029 65be bfb6 c9cb
A protocol from better times

- An ancient protocol
  - People were friendly and trustworthy
  - Internet was a warm and fuzzy place
- *DNS is a protocol from admins for admins*
  - Main assumption: Computers do not lie
  - Idea: A hierarchical distributed database
- Store locally, read globally
Playground to extend

• DNS **works**, so use is as a container
  • http://tools.ietf.org/wg/dnsext/
• DNS **scales**, so push a lot of data in
  • in-addr.arpa
• DNS can be **misused** as a catchword repository: www.catchword.com
• DNS may have **multiple roots**, so introduce private name spaces
Playground to manipulate

• Push all initial requests to a payment site
• Prevent requests to *bad* sites
• Offer own search engine for NXDOMAIN
• Geolocation for efficient content delivery
• Geolocation for lawful content selection
• Provide different software updates
• Prevent worm updates
trustroute +trace

- Modelling real world data as DNS records
- Transferring data into DNS primary server
- Transferring data into DNS secondaries
- Updating meta data in parent zone
- Delivering data to recursive servers
- Processing by resolver code
- Providing structures to applications
- Interpreting data by users
Securing the data flow

• Two possible design goals:
  • Detect manipulation
  • Prevent wire-tapping

• Facing typical problems
  • The compatibility hydra
  • Partial roll-out
  • Satellite networks

• Still designed by admins: NSEC(3)
DNS SECurity

- Trust the primary name server data
  - Signed by the zone-c
- A framework to verify integrity
  - Signature chains up to a trust anchor
- In band key management
  - DS records in parent zone (but glue!)
- Supports caching as well as offloading
- Provides peer authentication
Trust anchor management

• The root **is** signed
• In band key roll-overs: RFC 5011
• Fill the gaps (parent zone not signed)
  • Manual trust anchors: Edit files on disk
  • Trust Anchor Repositories: Look aside zones
    DS do.main => DLV do.main.dlv.pro.vi.der
• Question: Precedence of sources?
The last mile

- In an ideal world, apps use a new API
  - Error messages might become helpful
  - Validation errors are SERVFAIL
- Resolver offloading
  - Provide validated data with AD
  - Allow validator chaining with CD
  - Question: Provide bogus data at all?
- Attacks on the last mile even for LEAs
Finally gain benefits

- **DNSSEC adds trust to DNS**
- Use DNS as a hierarchical distributed DB
  - Manage your SSHFPs centrally
  - Manage your CERTs distributed
  - Manage your OpenPGP keys distributed
- Do not deliver poisoned data to clients
  - Validate late, validate centrally
Further Consequences

• Current practice for Intranets
  • Build a separate network using site specific names and numbers
  • Provide application layer gateways, NAT, Split-DNS, and VPN for non-local access
  • Hide internal structure
  • Statically map necessary services (Firewall)
  • Provide local “root” services (Active Directory)
The IPv6 impact

- IPv6 provides **public, globally routable IPs**
  - Clients do IPv6 automatically (even tunnel)
- IPv6 provides **end-to-end communication**
- IPv6 is **not designed to be translated**
- Future protocols rely on **direct channels**
  - Web 2.0: Numerous bits from different servers
  - Client to client communication
  - Shortest routing for “quality enhancements”
The DNSSEC impact

• Validation chain from a **well-known key**
  • Clients may have the key hardcoded
• **Only one root** possible
  • *No local names*
• Prevents rdata and NXDOMAIN rewriting
  • **Consistent** external and internal view
• Enterprise DNS rely on DNSSEC from everywhere (*DirectAccess, SSH, _tcp …*)
The horrible mobile client

• Public mobile networks are everywhere
• Mobile clients
  • Important status symbols
  • Roam in and out quickly
  • Always on: Cloud services
  • Can’t be configured
• IPv6
  • Exposes internal DNS servers
  • Create mobile peer-to-peer networks
Future (Intra)Nets
Modern intranets

• **Accept** consistency requirement
  • Local WLAN *and* mobile networks
  • REST web applications instead of VPN
• Secure the services, not the networks
• Secure the data, not the servers (cloud)
• Authenticate the user, not the computer
• Use DNS as trustworthy resource
• Always use direct communication
Conclusion

• IPv6 and DNSSEC dramatically change the design of modern networks
  • Information hiding policies do not work
  • Centralized policy enforcement unusable
• Concentrate on benefits
  • Build stable, globally routable networks
  • Enforce data security at the data level
  • Trust the people, not the devices
Did you sign your zones?

Why not?